

THE INVENTION CLAIMED IS:

1 1. A method of allocating memory requests in a
2 physical memory space defined between a first boundary and a
3 second boundary, the method comprising the steps of:
4 allocating non-pinned memory requests from
5 the first boundary; and
6 allocating pinned memory requests from the
7 second boundary.

2. The method of claim 1, wherein a non-pinned region is provided from the first boundary toward the second boundary, a pinned region is provided from the second boundary toward the first boundary and a free region is defined between the non-pinned region and the pinned region, and further comprising:

- receiving a pinned memory request;
- determining whether a hole that accommodates the received pinned memory request exists in the pinned region;
- if such a hole exists, allocating the received pinned memory request to the hole; and
- if such a hole does not exist, allocating the received pinned memory request in the free region adjacent the pinned region.

1 3. The method of claim 2, wherein if more than
2 one such hole exists, the received pinned memory request is
3 allocated to a smallest one of such holes.

1 4. The method of claim 2, further comprising
2 operating a background process to compact the non-pinned

3 region by moving at least one non-pinned memory block from
4 an end of the non-pinned region adjacent the free region to
5 fill at least one hole in the non-pinned region.

1 5. A method of allocating a memory request in a
2 physical memory space defined between a first boundary and a
3 second boundary, a non-pinned region being provided from the
4 first boundary toward the second boundary and a pinned
5 region being provided from the second boundary toward the
6 first boundary and a free region being defined between the
7 non-pinned region and the pinned region, the method
8 comprising the steps of:

9 receiving a pinned memory request;

10 determining that there is no hole in the
11 pinned region that accommodates the received pinned memory
12 request;

13 determining that the free region is not large
14 enough to accommodate the pinned memory request;

15 widening the free region by moving a non-
16 pinned memory block from an end of the non-pinned region
17 adjacent the free region to a hole in the pinned region; and

18 allocating the received pinned memory request
19 to the widened free region.

1 6. A method of allocating a memory request in a
2 physical memory space defined between a first boundary and a
3 second boundary, a non-pinned region being provided from the
4 first boundary toward the second boundary and a pinned
5 region being provided from the second boundary toward the
6 first boundary and a free region being defined between the

7 non-pinned region and the pinned region, the method
 8 comprising the steps of:
 9 receiving a pinned memory request;
 10 determining that there is a non-pinned memory
 11 block located in the pinned region that occupies a hole at
 12 least as large as the received pinned memory request;
 13 moving said non-pinned memory block located
 14 in the pinned region to free said hole that is at least as
 15 large as the received pinned memory request; and
 16 allocating the received pinned memory request
 17 to said freed hole that is at least as large as the received
 18 pinned memory request.

1 7. The method of claim 6, wherein the non-pinned
 2 memory block is moved to the free region adjacent the non-
 3 pinned region.

1 8. A digital memory apparatus, comprising:
 2 a memory component providing a physical
 3 memory space, the physical memory space defined between a
 4 first boundary and a second boundary; and
 5 control means for controlling allocation of
 6 the physical memory space, the control means being
 7 programmed to:
 8 allocate non-pinned memory requests from the
 9 first boundary; and
 10 allocate pinned memory requests from the
 11 second boundary.

1 9. The apparatus of claim 8, wherein a non-
 2 pinned region is provided from the first boundary toward the

3 second boundary, a pinned region is provided from the second
4 boundary toward the first boundary and a free region is
5 defined between the non-pinned region and the pinned region,
6 and the control means is further programmed to:

7 receive a pinned memory request;

8 determine whether a hole that accommodates
9 the received pinned memory request exists in the pinned
10 region;

11 if such a hole exists, allocate the received
12 pinned memory request to the hole; and

13 if such a hole does not exist, allocate the
14 received pinned memory request in the free region adjacent
15 the pinned region.

1 10. The apparatus of claim 9, wherein if more
2 than one such hole exists, the control means is programmed
3 to allocate the received pinned memory request to a smallest
4 one of such holes.

1 11. The apparatus of claim 9, wherein the control
2 means is further programmed to operate a background process
3 to compact the non-pinned region by moving at least one non-
4 pinned memory block from an end of the non-pinned region
5 adjacent the free region to fill at least one hole in the
6 non-pinned region.

1 12. A digital memory apparatus, comprising:
2 a memory component providing a physical
3 memory space, the physical memory space defined between a
4 first boundary and a second boundary, a non-pinned region
5 being provided from the first boundary toward the second

13 receive a pinned memory request;
 14 determine that there is a non-pinned
 15 memory block located in the pinned region that occupies a
 16 hole at least as large as the received pinned memory
 17 request;
 18 move said non-pinned memory block
 19 located in the pinned region to free said hole that is at
 20 least as large as the received pinned memory request; and
 21 allocate the received pinned memory
 22 request to said freed hole that is at least as large as the
 23 received pinned memory request.

1 14. The apparatus of claim 13, wherein the
 2 control means is programmed to move the non-pinned memory
 3 block to the free region adjacent the non-pinned region.

1 15. A computer program product comprising:
 2 a medium readable by a computer, the computer
 3 readable medium having computer program code adapted to:
 4 allocate non-pinned memory requests from
 5 a first boundary of a physical memory space defined between
 6 the first boundary and a second boundary; and
 7 allocate pinned memory requests from the
 8 second boundary of the physical memory space.

1 16. A method of handling memory allocation
 2 requests in a physical memory space, a non-pinned region
 3 being defined at a first end of the physical memory space
 4 and a pinned region being defined at a second end of the
 5 physical space, the second end being opposite to the first

6 end, a free region being defined between the non-pinned
 7 region and the pinned region, the method comprising:
 8 (i) responding to a non-pinned memory
 9 allocation request by allocating free space from the non-
 10 pinned region first, from the free region second and from
 11 the pinned region last; and
 12 (ii) responding to a pinned memory request
 13 only by one of: (a) allocating space from a hole in the
 14 pinned region or (b) allocating free space from the free
 15 region.

1 17. The method of claim 16, wherein step (i)
 2 includes allocating free space from the free region
 3 beginning from the non-pinned region toward the pinned
 4 region, and step (ii) includes allocating free space from
 5 the free region beginning from the pinned region toward the
 6 non-pinned region.

1 18. The method of claim 16, wherein step (ii)
 2 includes moving at least one non-pinned page from the hole
 3 in the pinned region.

1 19. The method of claim 16, wherein step (ii)
 2 includes widening the free region.

1 20. The method of claim 19, where the widening of
 2 the free region includes moving at least one non-pinned page
 from an end of the non-pinned region.